

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Penelitian yang dilakukan oleh (Prastya et al. 2021) dengan judul penelitian Sistem informasi Peternak Ayam broiler Berbasis Web. Pada penelitian ini broiler merupakan salah satu ras ayam yang paling produktif untuk produksi daging. Menurut Badan Pusat Statistik Indonesia, konsumsi daging ayam di Indonesia semakin meningkat dari tahun ke tahun. Peternak unggas broiler diharapkan dapat meningkatkan produksinya guna memenuhi kebutuhan konsumsi daging yang semakin meningkat[3].

Penelitian yang dilakukan oleh (Fauzi and Murti 2022) dengan judul penelitian Perancangan Sistem Informasi Penjualan Ayam Negri Berbasis Aplikasi Android Di Cv.Suyadi Broiler Pada penelitian ini pembuatan aplikasi ini menggunakan android studio dan menggunakan metode pengembangan sistem waterfaall sedangkan dalam pembuatan database dengan menggunakan MySQL[4].

Penelitian yang dilakukan oleh (Aisyah, Mihrani, and Nur 2020) dengan judul penelitian Analisis distribusi dan margin pemasaran ayam broiler dengan pakan herbal di Kecamatan Mandalle Kabupaten Pangkep. Pada penelitian ini Tataniaga mendukung peternak dalam mendistribusikan produk peternakan ke konsumen. Penelitian ini dilakukan untuk mengetahui format pemasaran ayam broiler dengan pakan herbal. Mengukur dan menganalisis keuntungan dan margin yang dicapai lembaga pemasaran

ayam broiler [5].

Penelitian yang dilakukan oleh (Wulandari, Putri, and Hendro Wahyudiono 2022) dengan judul penelitian Sistem Informasi Pengolahan Data Peternakan Ayam Kampung Pada CV. Dua Saudara Berbasis Web Mobile Pada penelitian ini membuat sebuah sistem informasi pengolahan data peternakan ayam menggunakan Bootstrap, PHP , dan MySql sebagai database[6].

Penelitian yang dilakukan oleh (Arifin and Budiono 2019) dengan judul penelitian Sistem Informasi Administrasi Dan Pemesanan Kebutuhan Ayam Pada Pt Muria Jaya Raya Kudus Pada penelitian ini perkembangan ilmu pengetahuan dan teknologi berjalan dengan sangat pesat. Semua ini dikembangkan dengan tujuan membantu masyarakat menjalankan tugas dan minatnya dengan mudah. Seperti proses manajemen registrasi bergabung dalam kemitraan PT Muria Jaya Raya Kudus yang menangani penyediaan bibit ayam broiler, Dalam pembuatan aplikasi “Sistem Informasi Manajemen dan Pemesanan” yang bertujuan untuk mengatasi permasalahan yang saat ini dihadapi oleh peternakan ayam boiler, dll. Ayam dan obat-obatan tersebut Digunakan sebagai vaksin[7].

2.2. Landasan Teori

2.2.1. Pengertian Ayam Broiler

Ayam broiler merupakan jenis ayam pedaging unggul dan sudah banyak ditenakan di Indonesia. Broiler merupakan ayam yang diciptakan dari perkawinan silang, seleksi, dan rekayasa genetik. Dewasa ini, usaha peternakan broiler semakin berkembang seiring dengan perkembangan genetik ayam. (Mery Christiana Simanjuntak, 2018) Sudah banyak ayam broiler yang pertumbuhannya sangat cepat dengan Feed Conversion Ratio (FCR) yang rendah[8].

2.2.2. Peternak

Peternakan merupakan salah satu dari lima subsektor pertanian. Peternakan adalah kegiatan memelihara hewan ternak untuk dibudidayakan dan mendapatkan keuntungan dari kegiatan tersebut. Subsektor peternakan terbagi menjadi ternak besar, yaitu sapi, kerbau, dan kuda, dan ternak kecil yang terdiri dari kambing, domba, dan babi serta unggas (ayam, itik, dan burung puyuh)[9].

2.2.3. Boosttrap

Bootstrap adalah kerangka kerja CSS yang sumber terbuka dan bebas untuk merancang situs web dan aplikasi web. Kerangka kerja ini berisi templat desain berbasis HTML dan CSS untuk tipografi, formulir, tombol, navigasi, dan komponen antarmuka lainnya, serta juga ekstensi opsional JavaScript. Tidak seperti kebanyakan kerangka

kerja web lainnya, kerangka kerja ini hanya fokus pada pengembangan front-end saja[10]

2.2.4. Visual Studio Code

Visual Studio Code (Vc Code) ini adalah sebuah teks editor ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux*, *Mac* dan *Windows*. Teks editor ini secara langsung mendukung bahasa pemrograman *JavaScript*, *Typescript* dan *Node.js*, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang melalui *marketplace VS Code* (seperti *C++*, *C#*, *Python*, *Go*, *Java* dan beberapa bahasa pemrograman lainnya). Banyak sekali fitur-fitur yang disediakan oleh *VS Code*, diantaranya *Intellisense*, *Git Intregation*, *Debugging* dan fitur ekstensi yang menambah kemampuan teks editor. Fitur-fitur tersebut akan terus bertambah seiring dengan bertambahnya versi *VS Code*. Pembaruan versi *VS Code* ini juga dilakukan berkala setiap bulan, dan inilah yang membedakan *VS Code* dengan teks editor yang lain[11].

2.2.5. MYSQL

MySQL atau kepanjangan dari My Structured Structured Query Language. MySQL merupakan bahasa terstruktur yang khusus digunakan untuk mengolah database. MySQL pertama kali didefinisikan oleh American National Standards Institute (ANSI) pada tahun 1986. MySQL adalah sebuah sistem manajemen database

yang bersifat open source MySQL dapat digunakan untuk membuat dan mengelola database beserta isinya[12].

2.2.6. DataBase

Basisdata (database) adalah suatu alat yang digunakan untuk menyimpan informasi, mengambil informasi kapanpun dibutuhkan, dan mengatur informasi yang tersimpan. Jika menggambarkan lemari file merupakan suatu *basisdata*.

DBMS (Database Management System) merupakan suatu perangkat lunak yang digunakan untuk membuat, memelihara, mengontrol dan mengakses *basisdata* secara praktis dan efisien. Sedangkan *RDBMS* merupakan salah satu *DBMS* yang mendukung adanya relasi atau hubungan[13].

2.2.7. XAMPP

XAMPP dikembangkan dari sebuah tim proyek bernama Apache Friends, yang terdiri dari Tim Inti (Core Team), Tim Pengembang (Development Team) & Tim Dukungan (Support Team). XAMPP adalah singkatan yang masing-masing hurufnya adalah: X=Program ini dapat dijalankan di banyak sistem operasi seperti: Windows, Linux, Mac OS, dan Solaris. A=Apache, merupakan aplikasi WebServer. Tugas utama Apache adalah menghasilkan halaman web yang benar kepada user berdasarkan kode PHP yang dituliskan oleh pembuat halaman web[14].

2.2.8. PHP

PHP (Hypertext Preprocessor) Pertama kali ditemukan pada 1995 oleh seorang *Software Developer* bernama Rasmus Lerdorf. Ide awal *PHP* adalah ketika itu Rasmus ingin mengetahui jumlah pengunjung yang membaca *resumeonline*. *Script* yang dikembangkan baru dapat melakukan dua pekerjaan, yakni merekam informasi pengunjung dan menampilkan jumlah pengunjung dari suatu *website*. Dan sampai sekarang kedua tugas tersebut masih tetap populer digunakan di dunia web saat ini. Kemudian, dari situ banyak orang di milis mendiskusikan *script* buatan Rasmus Lerdorf, hingga akhirnya rasmus mulai membuat sebuah *tool/script* bernama *PHP*, *PHP* dibangun dari *scripts* yang ditulis secara *plaintext*. *PHP Interpreter* adalah bagian dari perangkat lunak yang ada pada *Web Server*, yang membaca file tersebut dan mengartikannya, memberikan keluaran *HTML* dan petunjuk mengenai bagaimana perilaku yang ada maupun menginterpretasikan masukan dari pengguna[15].

2.2.9. HTML

HTML (Hypertext Markup Language) adalah suatu cara memberikan tanda yang memberikan perintah kepada *browser* bagaimana suatu teks terstruktur. *HTML* memberikan perintah kepada *browser* bagaimana struktur dari dokumen, bagaimana *heading*-nya, bagaimana paragrafnya, bagaimana suatu teks akan ditampilkan dan lainnya. Dengan informasi yang diberikan, *browser*

dibangun dengan perintah dasar bagaimana menampilkan setiap elemen yang ada[16].

2.2.10. CSS

CSS (Cascading Style Sheets) adalah aturan untuk mengatur beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. CSS bukan merupakan bahasa pemrograman. Sama halnya *styles*, misalnya *heading*, *subbab*, *footer*, *images*, dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (file). Pada umumnya CSS dipakai untuk memformat tampilan halaman web yang dibuat dengan bahasa HTML dan XHTML[16].

2.2.11. UML

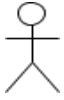
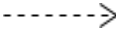
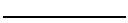
Unified Modeling Language atau lebih sering dikenal dengan sebutan *UML*, adalah salah satu metode dalam teknik rekayasa perangkat lunak yang digunakan untuk menggambarkan alur dan cara kerja sistem, fungsi, tujuan dan mekanisme kontrol sistem tersebut. Dalam teknik rekayasa perangkat lunak bidang analisis dan perancangan sistem informasi, saat ini lebih banyak menggunakan gabungan dari konsep pemrograman berorientasi objek dengan teknik pembuatan perangkat lunak, dimana suatu sistem dilihat sebagai objek tersendiri yang sudah mencakup data dan proses atau dapat bekerja secara mandiri dalam satu set sistem (*package*)

1. Use Case Diagram

Use Case diagram digunakan untuk mengkomunikasikan

interaksi manusia (actor) dengan apa yang bisa dilakukan oleh sistem. Sebuah *Use Case* dapat mewakili beberapa jalur interaksi manusia dengan sistem dan setiap jalur disebut sebagai skenario [17]. Simbol *Use Diagram* dapat dilihat seperti pada Tabel 2.1.

Tabel 2. 1 Simbol Use Case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.

NO	GAMBAR	NAMA	KETERANGAN
			
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		Note	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

2. Diagram Activity (Activity Diagram)

Activity diagram adalah diagram yang menggambarkan alur kerja atau aktivitas suatu sistem dalam perangkat lunak. Simbol *Activity Diagram* dapat dilihat seperti pada Tabel 2.2.

Tabel 2. 2 Simbol Activity Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi

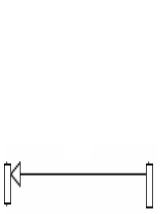
NO	GAMBAR	NAMA	KETERANGAN
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

3. Diagram Urutan (Sequence Diagram)

Sequence diagram mengilustrasikan objek-objek yang terdapat pada *Use Case* dan menggambarkan arus pesan antara satu sama lain pada Objek *Use Case*. *Sequence Diagram* bersifat dinamis dan lebih banyak menampilkan aktifitas objek berdasarkan urutan waktu[18]. Simbol *Sequence Diagram* dapat dilihat seperti pada Tabel 2.3.

Tabel 2. 3 Simbol Squence Diagram

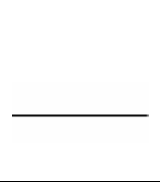
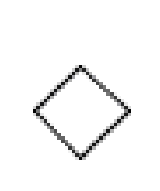
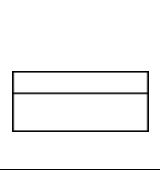
NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
			Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

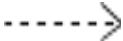
NO	GAMBAR	NAMA	KETERANGAN
2		<i>Message</i>	
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

4. Class Diagram

Class Diagram adalah model statis yang mendukung tampilan data dan informasi dari keseluruhan sistem. Penggunaan *Class Diagram* dikaitkan dengan struktur basis data sistem atau dapat menggantikan ERD pada proses penggambaran diagram rekayasa perangkat lunak yang konvensional[19]. Simbol *Class Diagram* dapat dilihat seperti pada Tabel 2.4.

Tabel 2. 4 Simbol Class Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.

NO	GAMBAR	NAMA	KETERANGAN
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya
8		Delay Symbol	Adalah simbol yang digunakan untuk menunjukkan proses delay (menunggu) yang perlu dilakukan. Seperti menunggu surat untuk diarsipkan dll.